正例からのパターン上の決定木の帰納推論 (計算モデルとアルゴリズム)

寺田, 幹治; 向内, 康人; 佐藤, 優子

数理解析研究所講究録
1093 巻 1999 年 111-116

正例からのパターン上の決定木の帰納推論

寺田 幹治* 向内 康人† 佐藤 優子†
Mikiharu Terada Yasuhito Mukouchi Masako Sato
* 大阪府立大学大学院 理学系研究科
† 大阪府立大学 総合科学部

概 要: 正則パターン上の決定木の学習の問題は，DNA 配列からのモチーフの発見というゲノム情報科学の観点から PAC 学習の枠組みを用いて研究されてきた．本稿では，この問題を正例からの極限同定という枠組みで論じている．

パターンとは，定数記号と変数記号からなる文字列であり，パターン上の決定木は，内部ノードのラベルがパターンで，葉のラベルが 0 または 1 の決定木である．本稿では，各パターンに対して，co-パターンと呼ばれる文字列を導入し，その意味をパターン言語の補言語のある部分集合として定義する．その解釈の下で，パターン上の決定木の意味を定義し，深さが高々 n である決定木で定義される言語の族が正例から推論可能であることを示す．特に，正則パターン上の深さが 1 である決定木で定義される言語の族に関しては，正例から多項式時間の更新で推論するアルゴリズムを与える．

1. はじめに

DNA 配列の中の機能領域からモチーフを発見することは，分子生物学における重要な問題の一つである．Arikawa et al.[3] や Miyano[6] 等は，計算学習のパラダイムの一つである「PAC 学習」の枠組みを用いてこの問題の定式化を行い，その理論的研究から学習システムの開発及び計算機実験まで幅広い研究を行っている．DNA 配列からのモチーフを表現するために採用されたのは，正則パターン上の決定木である．また，学習システムに提示されるのは，DNA 配列の正の事例及び負の事例である．本稿では，以下に述べる「極限同定」に基づく帰納推論の枠組みを用いて，正の事例からパターン上の決定木を極限同定する問題を考える．

パターンとは，アルファベット Σ に含まれる定数記号と変数からなる文字列である．各変数が高々1 回しか出現しないパターンは**正則パターン**と呼ばれる．**パターン上の決定木** T は，内部ノードのラベルがパターンで，葉のラベルが 1 または 0 の決定木である．Σ 上の文字列 w が与えられると，パターンをラベルとするノードでは w がそのパターン言語に含まれるか否かに応じて左または右に分岐し，根から葉へ至る．葉のラベルが 1 ならば，文字列 w は受理され，0 ならば受理されない．決定木 T が定義する言語は，T で受理される文字列の集合である．したがって，決定木で定義される言語は，パターン言語やそれらの補言語に対して，高々有限回の積 (共通部分) 演算や和演算を施して得られる言語である．

パターン言語の族は，正例から帰納推論可能な言語族として Angluin[2] によって導入された．正例からの帰納推論とは，与えられた正の事例からそれらを説明する一般的な規則・表現 (例えば，パターンやパターン上の決定木等) を推測する過程である．本稿では，Gold[5] の極限同定と呼ばれる推論の成功基準の下で，上記のパターン上の決定木の正例からの帰納推論を考える．

言語演算の観点から推論可能性を扱った研究に，Wright[12] による「有限の弾力性」という概念がある．Wright は，有限の弾力性をもつ言語族が正例から推論可能であり，またこの性質が和演算に関して閉じていることを示した．また，パターン言語族が有限の弾力性をもつことを示した．Moriyama&Sato[7] は，有限の弾力性が積演算や連接演算等に関しても

閉じていることを示した．演算の閉包性に関するこれらの結果を用いると，もし，パターン言語とそれらの補言語からなる言語族が有限の弾力性をもつならば，深さが高々 d であるパターン上の決定木で定義される言語の族は，有限の弾力性を有することになり，正例から推論可能ということになる．

本稿では，このような観点から決定木で定義される言語を扱うため，各パターン p に対して，co-パターンと呼ばれる文字列 p^c を導入する．co-パターン p^c の意味 L を $L(p)^c$ の部分集合 $L(p^c) = \{w \in L(p^c) \mid |w| \geq \max(1, |p| - k)\}$ と定義する．ただし，k は固定された非負の整数で，$|p|$ は文字列の長さを表す．このような解釈の下で，言語族 $\mathcal{JPL}$ が有限の弾力性をもつことが示される．したがって，この言語族に和演算や積演算を高々 n 回施して得られる言語族 $\mathcal{JPL}(n)$ や深さが高々 d である決定木で定義される言語族 $\mathcal{TPL}_d$ は，共に有限の弾力性をもち正例から推論可能である．

正則パターン上の決定木に制限した場合もこれらの結果は成り立つ．本稿では，$k = 0$ の場合について，深さが 1 の決定木で定義される言語の族を Angluin の意味で正例から多項式時間の更新で推論可能とするアルゴリズムを与える．

2. 正例からの言語の帰納推論

アルファベット Σ 上のすべての文字列の集合を Σ^*で表し，空列を除く文字列の集合を Σ^+ で表す．また，有限集合 S の要素の個数を $\sharp S$ で表す．

文字列の無限列 $w_1, w_2, \cdots$ が言語 L の**正提示**であるとは，$\{w_n \mid n \geq 1\} = L$ が成り立つことである．文字列の無限列 $\sigma = w_1, w_2, \cdots$ の n 番目までの初期部分列を $\sigma[n]$ で表す．

言語族 $\mathcal{L} = L_1, L_2, \cdots$ が**帰納的言語の添字付き族** (indexed family of recursive languages) であるとは，次のような計算可能関数 $f : N \times \Sigma^* \to \{0,1\}$ が存在することをいう: $f(i, w) = 1$, if $w \in L_i$; 0, if $w \notin L_i$. ここで，添字 i は言語を定義するオートマトンや形式文法等を意味すると考えられるが，本稿ではパターンやパターン上の決定木等を意味する．

推論機械 M とは，次々に入力を要求し，次々に出力を生成する実行的な手続きのことであり，M が生成する出力を**推測**と呼ぶ．文字列の無限列 σ に対して，その有限列 $\sigma[n]$ が入力された後，M が生成する推測を $M(\sigma[n])$ で表す．推論機械 M が入力の列 σ に対して，添字 $g \in N$ に**収束する**とは，ある $m \in N$ が存在し，任意の $n \geq m$ に対して，$M(\sigma[n]) = g$ となることをいう．推論機械 M が言語 L を正例から**極限同定** (identification in the limit) する (または，単に**推論する**) とは，L の任意の正提示に対して，M が $L_i = L$ となる添字 i に収束することをいう．また，任意の言語 $L \in \mathcal{L}$ を正例から極限同定する推論機械 M が存在するとき，言語族 $\mathcal{L}$ は正例から**推論可能である**という．

推論機械 M が言語族 $\mathcal{L}$ を正例から推論し，各入力を受け取ってから推測を出力するまでに必要な時間がそれまでの入力の長さの和のある多項式で押えられる時，その推論機械 M は言語族 $\mathcal{L}$ を正例から**多項式時間 (の更新) で推論する**という．また，言語族 $\mathcal{L}$ を正例から多項式時間 (の更新) で推論する推論機械が存在する時，その言語族は，正例から多項式時間 (の更新) で推論可能であるという．

正例から推論可能であるための必要条件に，言語族に含まれる各言語の有限証拠集合の存在がある (Angluin[2])．

ここで，集合 $T_i \subseteq \Sigma^*$ が言語 L_i の**有限証拠集合**であるとは，(1) T_i は L_i の有限部分集合であり，(2) $T_i \subseteq L_j \subsetneq L_i$ となる添字 $j \in N$ が存在しないことをいう．

正例から推論可能であるための十分条件として，有限の厚さという概念がある．言語族 $\mathcal{L}$ が**有限の厚さ** (finite thickness) **をもつ**とは，任意の文字列 $w \in \Sigma^+$ に対して，$\sharp\{L \in \mathcal{L} \mid w \in L\}$ が有限であることをいう．

有限の厚さよりもっと一般的な十分条件が，Wright によってもたらされた (cf. Wright[12], Motoki et al.[8])．言語族 $\mathcal{L}$ が**有限の弾力性** (finite elasticity) **をもつ**とは，次の条件を満たす文字列の無限列 $w_0, w_1, \cdots \in \Sigma^*$ と言語の無限列 $L_1, L_2, \cdots \in \mathcal{L}$ が存在しないことをいう: 任意の $k \in N$ に対して，

$$(1) \quad \{w_0, w_1, \cdots, w_{k-1}\} \subseteq L_k, \qquad (2) \quad w_k \notin L_k.$$

Wright は，有限の弾力性が正例から推論可能であるための十分条件であるだけでなく，この性質が言語族の和演算 $\tilde{\cup}$ に関して閉じていることを示した．また，Moriyama&Sato[7] は，この性質が言語族の積演算 $\tilde{\cap}$ や連接演算等に関しても閉じていることを

示した．ここでは，本稿に関連する次の2つの族演算のみ記載する:

$$\mathcal{L}_1 \tilde{\cup} \mathcal{L}_2 = \{L_1 \cup L_2 \mid L_1 \in \mathcal{L}_1,\ L_2 \in \mathcal{L}_2\},$$
$$\mathcal{L}_1 \tilde{\cap} \mathcal{L}_2 = \{L_1 \cap L_2 \mid L_1 \in \mathcal{L}_1,\ L_2 \in \mathcal{L}_2\}.$$

言語族 $\mathcal{L}$ と正整数 n に対して，言語族に含まれる言語に高々 n 回の積演算 $\tilde{\cap}$ または和演算 $\tilde{\cup}$ を施して得られる言語族を $\mathcal{L}(n)$ で表す．

定理 2.1 (Wright[12], Moriyama&Sato[7])．言語族 $\mathcal{L}$ が有限の弾力性をもつならば，言語族 $\mathcal{L}(n)$ も有限の弾力性をもつ．

有限の弾力性は，言語族の通常の和演算 $\cup$ に関しては閉じているが，補演算に関しては必ずしも閉じていないことが示される (cf. Moriyama&Sato[7])．

3. 正例からのパターン上の決定木の帰納推論

Σ を少なくとも 2 個以上の文字を含むアルファベットとし，X を Σ と互いに素な可算集合とする．Σ, X の要素をそれぞれ**定数** (**記号**)，**変数** (**記号**) とよぶ．

パターンとは，$\Sigma \cup X$ 上の空でない文字列である．パターン p の長さを $|p|$ で表し，パターン全体の集合を $\mathcal{P}$ で表す．**代入**とは，すべての定数をそれ自身に写すパターンからパターンへの準同型写像のこととする．代入 θ によるパターン p の像を $p\theta$ で表す．本稿では，代入として，非消去代入，すなわち，$|x\theta| \geq 1$ $(x \in X)$ を満たす代入 θ に制限して考える．パターン p, q に対して，$p = q\theta$ となる代入 θ が存在するとき，p は q の**例化** (q は p の**汎化**) といい，$p \preceq q$ と表す．明らかに，$p \preceq q$ ならば，$|p| \geq |q|$ である．パターン p が生成する言語 $L(p)$ を，

$$L(p) = \{w \in \Sigma^+ \mid \exists\theta \text{ s.t. } w = p\theta\}$$

と定義する．明らかに，$w \in L(p)$ ならば，$|p| \leq |w|$ となる．言語 L を生成するパターンが存在するとき，L は**パターン言語**と呼ばれる．パターン言語の所属問題は，計算可能であるがNP完全である (Angluin[1])．また，$p \preceq q$ ならば，$L(p) \subseteq L(q)$ であるが，しかし，その逆は一般には成り立たない (Angluin[1])．パターン言語の族を $\mathcal{PL} = \{L(p) \mid p \in \mathcal{P}\}$ で表すことにする．

パターン上の決定木とは，葉の各ノードが 0 または 1 の，葉ではない各ノードがパターンのラベルをもつ決定木である．深さが高々 $n \in N$ であるパターン上の決定木の全体を $\mathcal{TP}_n$ で表す．与えられた文字列 $w \in \Sigma^*$ に対して，パターン p をラベルとしてもつノードでは，w がそのパターン言語 $L(p)$ に属するか否かがテストされ，その結果 (yes または no) により左または右に分岐していく．パターン上の決定木 T に対して，1 のラベルをもつ葉のノードに至る文字列の集合を**決定木 T によって定義される言語**といい，$L(T)$ で表す．また，言語族 $\mathcal{TPL}_n = \{L(T) \mid T \in \mathcal{TP}_n\}$ とする．図 1の決定木 T は，深さ 2 の決定木である．そして，T によって定義されるされる言語は，

$$L(T) = (L(xaby) \cap L(axbby)^c) \cup (L(xaby)^c \cap L(xayb))$$

である．

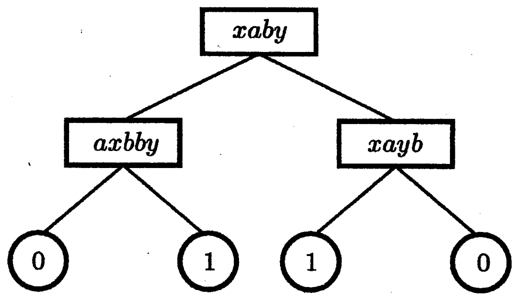

図 1: パターン上の決定木の例

この例でわかるように，パターン $p_1, p_2, \cdots, p_n$ をラベルにもつノードを経由して，1 のラベルをもつ葉のノードで受理される言語は，$L(p_i)$ またはその補言語 $L(p_i)^c$ の i についての共通部分であり，n は決定木の深さを越えない．そして，$L(T)$ は，そのような言語の集合和である．いま，言語族 $\mathcal{L} = \mathcal{PL} \cup \{L(p)^c \mid p \in \mathcal{P}\}$ とおくと，言語族 $\mathcal{TPL}_d$ は，$\mathcal{L}(n)$ の部分族である．ただし，$n = d2^{d-1}$ とする．もし，$\mathcal{L}(n)$ が正例から推論可能ならば，その部分族である $\mathcal{TPL}_d$ も正例から推論可能となる．本稿では，正例からパターン上の決定木を帰納推論する問題を考える．まず，パターン言語の補言語の問題からはじめよう．

明らかに，どんなパターン p に対しても，その補言語 $L(p)^c$ はパターン言語とはならない．ここでは，各パターン p に対して，p の co-パターンと呼ばれる文字列 p^c を導入する．co-パターン全体の集合を co-$\mathcal{P}$ で表し，$\mathcal{JP} = \mathcal{P} \cup$ co-$\mathcal{P}$ とおく．新しい記号列である co-パターン p^c の意味 (言語) をどう解釈するか．以下，co-パターン p^c の意味を補言語 $L(p)^c$ のある部分集合として定義し，パターンの意味と同じ記号 L を用いて $L(p^c)$ で表す．co-パターンの言語の全体を co-$\mathcal{PL}$ とし，$\mathcal{JPL} = \mathcal{PL} \cup$ co-$\mathcal{PL}$ とおく．

3.1. co-パターンの補言語としての解釈

各パターン p に対して，その co-パターンの意味を補言語 $L(p^c) = L(p)^c$ $(= \Sigma^* \setminus L(p))$ と解釈する．このとき，言語族 co-$\mathcal{PL}$ は有限の弾力性ももたない．実際，長さが単調に増加する文字列の無限列 $a, aa, aaa, \cdots$ および co-パターン言語の無限列 $L(aa^c), L(aaa^c), L(aaaa^c), \cdots$ はこの言語族が有限の弾力性をもたないことの例となる．しかしながら，この言語族は，正例から推論可能となることが，Shinohara[11] による「パターン言語族は，負例から推論可能である」という結果から直ちに導ける．

この解釈の下で次の結果が得られる．

定理 3.1. (i) 言語族 $\mathcal{JPL}$ $(= \mathcal{TPL}_1)$ は正例から推論可能である．

(ii) 言語族 $\mathcal{TPL}_2$ は正例から推論可能ではない．

この定理から，co-パターンの意味を補言語と解釈すると，$n > 1$ に対して，パターン上の深さが高々 n である決定木は正例から推論可能ではないことになる．

注) co-パターン p^c の解釈を $L(p^c) = \Sigma^+ \setminus L(p)$ とすると，上記の定理の証明と同様にして，$\mathcal{TPL}_1$ が正例から推論可能ではないことが示される．

3.2. 有限の厚さをもつ $\mathcal{JPL}$

ここでは，co-パターン p^c の意味として，次のような $L(p)^c$ の部分集合を考える:

$$L(p^c) = \{w \in L(p)^c \mid |w| \geq \max(1, |p| - k)\}.$$

ただし，k は任意に固定された非負整数とする．この定義から，以下の結果が成り立つ．

定理 3.2. 言語族 $\mathcal{JPL}$ は有限の厚さをもつ．

さて，co-パターンの意味を上記のように与えた場合の，パターン上の決定木 T で定義される言語の新しい意味 $L(T)$ を次のように与える．任意の文字列 w の入力に対して各ノードのラベルにあるパターン p と照合し，$w \in L(p)$, $w \in L(p^c)$ あるいは $w \notin L(p) \cup L(p^c)$ かに応じ，左，右への分岐あるいは停止の3つの選択肢がある．停止は，$w \notin L(T)$ を意味するものとする．停止となるのは，$|w| < |p| - k$ となる文字列であり，またそれらに限られる．したがって，パターン上の決定木 T に対して，途中で停止に陥る文字列の個数は高々有限個である．このような決定木の新しい意味 L の下で，深さが高々 d の決定木で定義される言語の族をあらためて $\mathcal{TPL}_d$ で表す．

正例からの決定木の推論に関しては次の定理がなりたつ．

定理 3.3. 任意の $n \in N$ に対して，言語族 $\mathcal{JPL}(n)$, $\mathcal{TPL}_n$ はいずれも有限の弾力性をもつ．したがって，これらの言語族は正例から推論可能である．

以下，言語族 $\mathcal{TPL}_d$ の基本となる言語族 $\mathcal{JPL}$ の有限証拠集合について論じる．

パターン p に対して，

$$S_p = \{w \in L(p) \mid |w| \leq |p| + k\},$$
$$S_{p^c} = \{w \in L(p^c) \mid |w| \leq |p|\}$$

と定義する．

定理 3.4. 言語族 $\mathcal{JPL}$ において，集合 S_p, S_{p^c} はそれぞれ $L(p), L(p^c)$ の有限証拠集合である．

4. 正則パターン上の決定木

本節では，正則パターン上の決定木で定義される言語について考察する．**正則パターン**とは，各変数が高々1回しか現れないパターンをいう．正則パターンで定義される言語は**正則パターン言語**と呼ばれる．正則パターンの全体および正則パターン言語の全体をそれぞれ，$\mathcal{RP}$ および $\mathcal{RPL}$ で表すことにする．

4.1. 正則パターン上の決定木の帰納推論

正則パターンに関しては，次の結果が得られている．

定理 4.1 (Shinohara[10], Mukouchi[9]). 正則パターン言語に関して，次の性質が成り立つ:

(i) 言語族 $\mathcal{RPL}$ における所属問題は，多項式時間で計算可能である．

(ii) $\sharp\Sigma \geq 3$ ならば，任意の正則パターン p, q に対して，$p \preceq q \Longleftrightarrow L(p) \subseteq L(q)$.

正則パターンの co-パターンからなる族を co-$\mathcal{RP}$ とし，$\mathcal{JRP} = \mathcal{RP} \cup$ co-$\mathcal{RP}$ とする．本節では，有限の厚さを持つことが示せた3.2節の co-正則パターンの解釈に従う．また，ここでは特に，$k = 0$ の場合について論じる．すなわち，正則パターン p に対して，

$$L(p^c) = \{w \in L(p)^c \mid |w| \geq |p|\}.$$

この解釈の下で，co-$\mathcal{RPL} = \{L(p^c) \mid p \in \mathcal{RP}\}$, $\mathcal{JRPL} = \mathcal{RPL} \cup$ co-$\mathcal{RPL}$ とおく．前節の結果は，正則パターンに対してもなりたつ．

co-パターン言語の定義から明らかに，$L(p) \subseteq L(q)$ ならば，$L(q^c) \subseteq L(p^c)$ であるが，逆は成り立たない．しかし，次の条件の下では，この不等式が成り立つ．

補題 4.2. 正則パターン p, q に対して，$L(p^c) \neq \phi$ かつ $L(p^c) \subseteq L(q^c)$ ならば，$|p| \geq |q|$ である．

4.2. 深さ1の決定木の効率的な学習アルゴリズム

本節では，正例からの正則パターン上の深さ1の決定木の効率的な学習アルゴリズムについて考察する．深さ1の決定木は，正則パターンかあるいはco-正則パターンである．

一般に，有限の弾力性をもつ言語族 $\mathcal{L}$ において，文字列の有限集合 $S \subseteq \Sigma^+$ を含む極小言語の1つを見つける問題が計算可能ならば，次のアルゴリズムIA は $\mathcal{L}$ に含まれる目標言語を推論することが知られている (cf. Arimura et al.[4])．ここで，$\mathrm{MINL}_{\mathcal{L}}(S)$ は，S の言語族 $\mathcal{L}$ における極小言語の添字を計算する手続きを表す．すなわち，$\mathrm{MINL}_{\mathcal{L}}(S) = i$ ならば，L_i は S を含み，かつ，$S \subseteq L_j \subsetneq L_i$ を満たす添字 $j \in N$ は存在しない．

Algorithm IA
入力: 文字列の無限列 $w_1, w_2, \cdots$;
出力: 言語の添字の無限列 $g_1, g_2, \cdots$;
begin
  $S := \phi$; $g_0 := -1$; $n := 1$;
  repeat
    read the next data w_n; $S := S \cup \{w_n\}$;
    if $w_n \notin L_{g_{n-1}}$ **then** $g_n := \mathrm{MINL}_{\mathcal{L}}(S)$
    else $g_n := g_{n-1}$;
    output g_n; $n := n + 1$
  forever
end.

定理 4.3 (Shinohara[10]). 正則パターン言語族 $\mathcal{RPL}$ において，文字列の空でない有限集合 $S \subseteq \Sigma^+$ の極小言語を生成する正則パターンを $O(l^2 m)$ の時間で計算する手続き MINL が存在する．ただし，$l = \max\{|w| \mid w \in S\}$, $m = \sharp S$ とする．

co-正則パターンの言語族において，極小言語の co-パターンを計算する手続きを次に与える．ここで，正則パターン p の j 番目の位置にある記号が定数 $a \in \Sigma$ であるとき，その定数を変数 x と置き換えて得られる正則パターン q を $q = p[x \leftarrow (j, a)]$ で表す．したがって，$q\{x := a\} = p$ が成り立つ．

Procedure co-MINL
入力: 文字列の空でない有限集合 $S \subseteq \Sigma^+$;
出力: co-正則パターン;
begin
  let n be the length of shortest strings in S;
  if $\Sigma^n \not\subseteq S$ **then begin**
    let $a_1 a_2 \cdots a_n$ ($a_i \in \Sigma$) be an arbitrary
      string not in S;
    $p_1 := a_1 a_2 \cdots a_n$;
    for $i := 1$ **to** n **do begin**
      $q := p_i[x_i \leftarrow (i, a_i)]$;
      if $S \subseteq L(q^c)$ **then** $p_{i+1} := q$
      else $p_{i+1} := p_i$
    end;
    $p := p_{n+1}$
  end
  else begin
    let $b_1 b_2 \cdots b_{n-1}$ be an arbitrary
      string in Σ^{n-1};
    $p := b_1 b_2 \cdots b_{n-1}$
  end;
  output p^c
end

補題 4.4. 文字列の空でない有限集合 $S \subseteq \Sigma^+$ に対する手続き co-MINL の出力を p^c とすると，$L(p^c)$ は co-$\mathcal{RPL}$ における S の極小言語である．

補題 4.5. 文字列の空でない有限集合 $S \subseteq \Sigma^+$ に対して，co-MINL(S) は，$O(l^2 m)$ の時間で計算可能

である．ただし，$l = \max\{|w| \mid w \in S\}$, $m = \sharp S$ とする．

手続き co-MINL は，入力された有限集合 S の極小言語を co-$\mathcal{RPL}$ の中で探索した．以下，$\mathcal{JRPL}$ における S の極小言語を探索し，その記述である正則パターンまたは co-正則パターンを出力する手続きを与える．

定理 4.6. $\sharp\Sigma \geq 3$ とする．文字列の空でない有限集合 $S \subseteq \Sigma^+$ に対する次の手続きの出力 JMINL(S) を $\pi \in \mathcal{JRP}$ とすると，言語 $L(\pi)$ は言語族 $\mathcal{JRPL}$ における S の極小言語である．

Procedure JMINL
入力: 文字列の空でない有限集合 $S \subseteq \Sigma^+$;
出力: 正則パターンまたは co-正則パターン;
begin
  let n be the length of shortest strings in S;
  if $\Sigma^n \subseteq S$ **then** $p := x_1x_2\cdots x_n$ **else begin**
    $p :=$ MINL(S);
    if $p = x_1x_2\cdots x_n$ **then** $p :=$ co-MINL(S)
  end
end

定理4.6より，次の結果が得られる．

定理 4.7. $\sharp\Sigma \geq 3$ とする．正則パターン上の深さ 1 の決定木は，正例から多項式時間の更新で推論可能である．

参考文献

[1] D. Angluin, *Finding patterns common to a set of strings*, in Proceedings of the 11th Annual Symposium on Theory of Computing, (1979) 130–141.

[2] D. Angluin, *Inductive inference of formal languages from positive data*, Information and Control, **45** (1980) 117–135.

[3] S. Arikawa, S. Kuhara, S. Miyano, Y. Mukouchi, Y. Shinohara, and T. Shinihara, *A machine discovery from amino acid sequences by decision trees over regular patterns*, New Generation Computing, **11**(3,4) (1993) 361–375.

[4] H. Arimura, T. Shinohara and S. Otsuki, *Finding minimal generalizations of pattern languages and its application to inductive inference from positive data*, in Proceedings of the 11th Symposium on Theoretical Aspects Computer Science, Lecture Notes in Computer Science, **775** (1994) 646–660.

[5] E.M. Gold, *Language identification in the limit*, Information and Control, **10** (1967) 447–474.

[6] S. Miyano, *Learning theory towords Genome Informatics*, in Proceedings of the 4th Workshop on Algorithmic Learning Theory, Lecture Notes in Artificial Intelligence, **744** (1993) 19–36.

[7] T. Moriyama and M. Sato, *Properties of language classes with finite elasticity*, IEICE Transactions on Information and Systems, **E78-D**(5) (1995) 532–538.

[8] T. Motoki and T. Shinohara, *The correct definition of finite elasticity: corrigendum to identification of unions*, in Proceedings of the 4th Annual ACM Workshop on Computational Learning Theory, (1991) 375–375.

[9] Y. Mukouchi, *Characterization of pattern languages*, in Proceedings of the 2nd Workshop on Alogrithmic Learning Theory, (1991) 93–104.

[10] T. Shinohara, *Polynomial time inference of pattern languages and its applications*, in Preceedings of the 7th IBM Symposium on Mathematical Foundations of Computer Science, (1982) 191–209.

[11] T. Shinohara, *Inductive inference from negative data*, Bulletine of Informatics and Cybernetics **21**(3, 4) (1985) 67–70.

[12] K. Wright, *Identification of unions of language drawn from an identifiable class*, in Proceedings of the 2nd Workshop on Computational Learning Theory, (1989) 328–333.